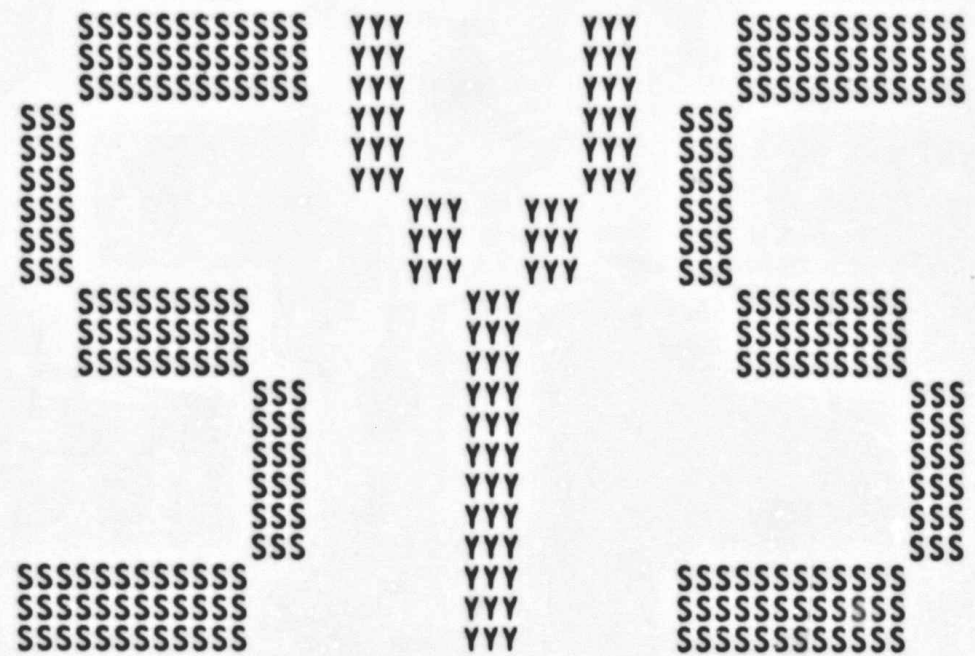




[illegible]

(1)	51	HISTORY	; DETAILED
(1)	63	DECLARATIONS	

```
0000 1 .TITLE CVT_ATB - ASCII NUMBER TO BINARY CONVERSION
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6
0000 7 *
0000 8 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 9 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 10 * ALL RIGHTS RESERVED.
0000 11 *
0000 12 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 13 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 14 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 15 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 16 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 17 * TRANSFERRED.
0000 18 *
0000 19 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 20 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 21 * CORPORATION.
0000 22 *
0000 23 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 24 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 25 *
0000 26 *****
0000 27
0000 28
0000 29 EQUATED SYMBOLS:
0000 30
00000004 0000 31 COUNT = 4 ; COUNT ARG IN ARG LIST
00000008 0000 32 STRING = 8 ; STRING ARG
0000000C 0000 33 RESULT = 12 ; RESULT ADDRESS
0000 34
0000 35 ++
0000 36
0000 37 FACILITY: SYSTEM LIBRARY
0000 38
0000 39 ABSTRACT:
0000 40
0000 41 THIS ROUTINE PERFORMS ASCII INTEGER TO BINARY CONVERSION IN
0000 42 DECIMAL, OCTAL, AND HEX RADIX; RADIX DETERMINED BY ENTRY POINT.
0000 43 AN OPTIONAL LEADING SIGN IS ACCEPTED.
0000 44
0000 45 ENVIRONMENT:
0000 46
0000 47 STAR NATIVE MODE PROCESSOR, ANY ACCESS LEVEL. NO SPECIAL
0000 48 INSTRUCTIONS OR SYSTEM SERVICES ARE USED. 9 LONGWORDS OF
0000 49 STACK SPACE NEEDED.
0000 50
0000 51 .SBTTL HISTORY ; DETAILED
0000 52
0000 53 AUTHOR: ANDREW C. GOLDSTEIN 26-JAN-78 16:48
0000 54
0000 55 MODIFIED BY:
0000 56
0000 57 V03-001 LJK0254 Lawrence J. Kenah 6-Dec-1983
```


CVT-ATB
V04=000

- ASCII NUMBER TO BINARY CONVERSION^{B 14}
HISTORY ; DETAILED

15-SEP-1984 23:57:53 VAX/VMS Macro V04-00
5-SEP-1984 03:41:03 [SYS.SRC]CVTATB.MAR;1

Page 2
(1)

0000 58 ;
0000 59 ;
0000 60 ;
0000 61 ;--

Change names of entry points from LIB\$CVT_xTB to FIL\$CVT_xTB
to avoid conflict with same names in RTL. Use .ENTRY directive
for register save mask.


```
0000 63      .SBTTL  DECLARATIONS
0000 64      :++
0000 65
0000 66      FUNCTIONAL DESCRIPTION:
0000 67
0000 68      THE ROUTINE WORKS IN THE OBVIOUS MANNER OF ALL CONVERSION ROUTINES:
0000 69      IT SCANS THE INPUT STRING AND CONVERTS EACH CHARACTER INTO ITS
0000 70      NUMERICAL EQUIVALENT AND CHECKS IT FOR LEGALITY AGAINST THE RADIX.
0000 71      THE NUMBER BEING ACCUMULATED IS THEN MULTIPLIED BY THE RADIX AND THE
0000 72      NEW DIGIT IS ADDED IN. NON-RADIX CHARACTERS IN THE INPUT STRING CAUSE
0000 73      AN ERROR RETURN. A SIGN IN OTHER THAN THE FIRST CHARACTER POSITION
0000 74      AND OVERFLOW FROM 32 BITS (UNSIGNED) ALSO CAUSE AN ERROR RETURN.
0000 75
0000 76      NOTE THAT THESE ROUTINES ARE IDENTICAL TO ROUTINES WITH THE SAME NAMES AND
0000 77      THE LIB$ PREFIX THAT EXIST IN THE RUN-TIME LIBRARY. THESE ROUTINES USE
0000 78      DIFFERENT PSECT NAMES AND DIFFERENT ENTRY POINT NAMES TO ALLOW THEM TO
0000 79      COEXIST IN IMAGES THAT LINK AGAINST BOTH STARLET.OLB AND SYS.STB.
0000 80
0000 81      CALLING SEQUENCE:
0000 82      CALL FIL$CVT_DTB (COUNT,STRING,RESULT) ; CONVERT DECIMAL TO BINARY
0000 83      CALL FIL$CVT_OTB (COUNT,STRING,RESULT) ; CONVERT OCTAL TO BINARY
0000 84      CALL FIL$CVT_HTB (COUNT,STRING,RESULT) ; CONVERT HEX TO BINARY
0000 85
0000 86
0000 87      INPUT PARAMETERS:
0000 88      COUNT:  BYTE COUNT OF INPUT STRING
0000 89      STRING: ADDRESS OF INPUT STRING
0000 90
0000 91      IMPLICIT INPUTS:
0000 92      NONE
0000 93
0000 94      OUTPUT PARAMETERS:
0000 95      RESULT: ADDRESS TO STORE LONGWORD RESULT
0000 96
0000 97      IMPLICIT OUTPUTS:
0000 98      NONE
0000 99
0000 100     COMPLETION CODES:
0000 101     R0 = 1  SUCCESSFUL CONVERSION
0000 102     = 0  ILLEGAL CHARACTER OR ZERO BYTE COUNT
0000 103
0000 104     SIDE EFFECTS:
0000 105     NONE
0000 106
0000 107     --
0000 108
0000 109     .ENABL  LSB
0000 110
00000000 111     .PSECT  YCVTATB,WRT
0000 112
0000 113     BASE AND VALUE TABLES TO CONVERT DIGITS INTO THEIR NUMERICAL VALUES
0000 114
30 41 61 0000 115 BASETAB:      .BYTE  ^A'a',    ^A'A',    ^A'0'
39 5A 7A 0003 116 TOPTAB:      .BYTE  ^A'z',    ^A'Z',    ^A'9'
30 37 57 0006 117 VALTAB:      .BYTE  ^A'a'-10, ^A'A'-10, ^A'0'-00
0009 118
0009 119
```



```
0009 120 .ENTRY - ; ENTRY FOR DECIMAL CONVERT
0009 121 FIL$CVT_DTB,-
52 0A 003C 0009 122 ^M<R2,R3,R4,R5> ; SAVE REGISTERS 2-5
OC 11 000B 123 #10,R2 ; R2 = RADIX
000E 124 10$
0010 125
0010 126 .ENTRY - ; ENTRY FOR OCTAL CONVERT
003C 0010 127 FIL$CVT_OTB,-
52 08 0012 128 ^M<R2,R3,R4,R5> ; SAVE REGISTERS 2-5
05 11 0012 129 #8,R2 ; R2 = RADIX
0015 130 10$
0017 131
0017 132 .ENTRY - ; ENTRY FOR HEXADECIMAL CONVERT
003C 0017 133 FIL$CVT_HTB,-
52 10 0019 134 ^M<R2,R3,R4,R5> ; SAVE REGISTERS 2-5
0019 135 #16,R2 ; R2 = RADIX
001C 136
50 7C 001C 137 10$: CLRQ R0 ; ZERO INITIAL NUMBER
53 7C 001E 138 CLRQ R3 ; AND STRING POINTER & SIGN
53 D7 0020 139 DECL R3 ; BACK OFF COUNT
40 11 0022 140 BRB 70$ ; AND ENTER PRE-TESTED LOOP
0024 141
55 08 BC43 9A 0024 142 20$: MOVZBL @STRING(AP)[R3],R5 ; GET NEXT CHARACTER
53 D5 0029 143 TSTL R3 ; SEE IF THIS IS FIRST CHARACTER
0E 12 002B 144 BNEQ 30$ ; SKIP SIGN CHECK IF NOT
2B 55 91 002D 145 CMPB R5,#^A'+ ; CHECK FOR PLUS
32 13 0030 146 BEQL 70$
2D 55 91 0032 147 CMPB R5,#^A'- ; CHECK FOR MINUS
04 12 0035 148 BNEQ 30$ ; BRANCH IF NOT
54 D6 0037 149 INCL R4 ; SET NEGATIVE FLAG
29 11 0039 150 BRB 70$ ; OK - LOOP FOR NEXT CHARACTER
003B 151
51 D4 003B 152 30$: CLRL R1 ; INIT TABLE INDEX
BE AF41 55 91 003D 153 40$: CMPB R5,BASETAB[R1] ; CHECK DIGIT AGAINST BASE CHARACTER
07 1F 0042 154 BLSSU 50$ ; BRANCH IF DIGIT IS LESS
BA AF41 55 91 0044 155 CMPB R5,TOPTAB[R1] ; CHECK DIGIT AGAINST TOP CHARACTER
06 1B 0049 156 BLEQU 60$ ; BRANCH IF LESS - VALID
EE 51 03 F2 004B 157 50$: AOBLS #3,R1,40$ ; NOT THIS RANGE - TRY NEXT LOWER BASE
26 11 004F 158 BRB 90$ ; NOT A LEGAL DIGIT
0051 159
55 B1 AF41 82 0051 160 60$: SUBB VALTAB[R1],R5 ; COMPUTE NUMERIC VALUE (NOTE - ALWAYS +)
52 55 D1 0056 161 CMPL R5,R2 ; CHECK DIGIT AGAINST RADIX
1C 1E 0059 162 BGEQU 90$ ; TOUGH LUCK .....
50 55 50 52 7A 005B 163 EMUL R2,R0,R5,R0 ; SCALE VALUE BY RADIX AND ADD DIGIT
51 D5 0060 164 TSTL R1 ; CHECK FOR OVERFLOW
13 12 0062 165 BNEQ 90$
0064 166
BB 53 04 AC F2 0064 167 70$: AOBLS COUNT(AP),R3,20$ ; LOOP FOR DIGITS
0069 168
03 54 E9 0069 169 BLBC R4,80$ ; CHECK SIGN BIT
50 50 CE 006C 170 MNEGL R0,R0 ; MAKE NEGATIVE IF DESIRED
OC BC 50 D0 006F 171 80$: MOVL R0,@RESULT(AP) ; STORE RESULT
50 01 D0 0073 172 MOVL #1,R0 ; RETURN TRUE
04 0076 173 RET
0077 174
0077 175 ; TO HERE ON ANY BAD CHARACTER
0077 176 ;
```


CVT_ATB
V04=000

- ASCII NUMBER TO BINARY CONVERSION^{E 14}
DECLARATIONS

15-SEP-1984 23:57:53 VAX/VMS Macro V04-00
5-SEP-1984 03:41:03 [SYS.SRC]CVTATB.MAR;1

Page 5
(1)

```
50  D4 0077 177 90$: CLRL R0
    04 0079 178      RET
      007A 179      .DSABL LSB
      007A 180
      007A 181
      007A 182
      007A 183
      007A 184      .END
```

; RETURN FALSE

CVT_ATB
Symbol table

- ASCII NUMBER TO BINARY CONVERSION F 14

15-SEP-1984 23:57:53
5-SEP-1984 03:41:03

VAX/VMS Macro V04-00
[SYS.SRC]CVTATB.MAR;1

Page 6
(1)

BASETAB = 00000000 R 02
COUNT = 00000004
FIL\$CVT_DTB = 00000009 RG 02
FIL\$CVT_HTB = 00000017 RG 02
FIL\$CVT_OTB = 00000010 RG 02
OPS_ACB0 = 0000006F
OPS_ACBF = 0000004F
OPS_ACBG = 00004FFD
OPS_ACBH = 00006FFD
OPS_ADDD2 = 00000060
OPS_ADDD3 = 00000061
OPS_ADDD4 = 00000040
OPS_ADDD5 = 00000041
OPS_ADDG2 = 000040FD
OPS_ADDG3 = 000041FD
OPS_ADDH2 = 000060FD
OPS_ADDH3 = 000061FD
OPS_ADDP4 = 00000020
OPS_ADDP6 = 00000021
OPS_ASHP = 000000F8
OPS_CLRD = 0000007C
OPS_CLRF = 000000D4
OPS_CLRG = 0000007C
OPS_CLRH = 00007CFD
OPS_CMPD = 00000071
OPS_CMPF = 00000051
OPS_CMPG = 000051FD
OPS_CMPH = 000071FD
OPS_CMPP3 = 00000035
OPS_CMPP4 = 00000037
OPS_CRC = 0000000B
OPS_CVTBD = 0000006C
OPS_CVTBF = 0000004C
OPS_CVTBG = 00004CFD
OPS_CVTBH = 00006CFD
OPS_CVTDB = 00000068
OPS_CVTDF = 00000076
OPS_CVTDH = 000032FD
OPS_CVTDL = 0000006A
OPS_CVTDW = 00000069
OPS_CVTFB = 00000048
OPS_CVTFD = 00000056
OPS_CVTFG = 000099FD
OPS_CVTFH = 000098FD
OPS_CVTFL = 0000004A
OPS_CVTFW = 00000049
OPS_CVTGB = 000048FD
OPS_CVTGF = 000033FD
OPS_CVTGH = 000056FD
OPS_CVTGL = 00004AFD
OPS_CVTGW = 000049FD
OPS_CVTHB = 000068FD
OPS_CVTHD = 0000F7FD
OPS_CVTHF = 0000F6FD
OPS_CVTHG = 000076FD
OPS_CVTHL = 00006AFD
OPS_CVTHW = 000069FD

OPS_CVTLD = 0000006E
OPS_CVTLF = 0000004E
OPS_CVTLG = 00004EFD
OPS_CVTLH = 00006EFD
OPS_CVTLP = 000000F9
OPS_CVTPL = 00000036
OPS_CVTPS = 00000008
OPS_CVTPT = 00000024
OPS_CVTRDL = 0000006B
OPS_CVTRFL = 0000004B
OPS_CVTRGL = 000048FD
OPS_CVTRHL = 000068FD
OPS_CVTSP = 00000009
OPS_CVTTP = 00000026
OPS_CVTWD = 0000006D
OPS_CVTWF = 0000004D
OPS_CVTWG = 00004DFD
OPS_CVTWH = 00006DFD
OPS_DIVD2 = 00000066
OPS_DIVD3 = 00000067
OPS_DIVF2 = 00000046
OPS_DIVF3 = 00000047
OPS_DIVG2 = 000046FD
OPS_DIVG3 = 000047FD
OPS_DIVH2 = 000066FD
OPS_DIVH3 = 000067FD
OPS_DIVP = 00000027
OPS_EDITPC = 00000038
OPS_EMDD = 00000074
OPS_EMDDF = 00000054
OPS_EMDDG = 000054FD
OPS_EMDDH = 000074FD
OPS_MATCHC = 00000039
OPS_MNEGD = 00000072
OPS_MNEGF = 00000052
OPS_MNEGG = 000052FD
OPS_MNEGH = 000072FD
OPS_MOVD = 00000070
OPS_MOVF = 00000050
OPS_MOVG = 000050FD
OPS_MOVH = 000070FD
OPS_MOVP = 00000034
OPS_MOVTC = 0000002E
OPS_MOVTUC = 0000002F
OPS_MULD2 = 00000064
OPS_MULD3 = 00000065
OPS_MULF2 = 00000044
OPS_MULF3 = 00000045
OPS_MULG2 = 000044FD
OPS_MULG3 = 000045FD
OPS_MULH2 = 000064FD
OPS_MULH3 = 000065FD
OPS_MULP = 00000025
OPS_POLYD = 00000075
OPS_POLYF = 00000055
OPS_POLYG = 000055FD
OPS_POLYH = 000075FD

OPS_SCANC = 0000002A
OPS_SKPC = 0000003B
OPS_SPANC = 0000002B
OPS_SUBD2 = 00000062
OPS_SUBD3 = 00000063
OPS_SUBF2 = 00000042
OPS_SUBF3 = 00000043
OPS_SUBG2 = 000042FD
OPS_SUBG3 = 000043FD
OPS_SUBH2 = 000062FD
OPS_SUBH3 = 000063FD
OPS_SUBP4 = 00000022
OPS_SUBP6 = 00000023
OPS_TSTD = 00000073
OPS_TSTF = 00000053
OPS_TSTG = 000053FD
OPS_TSTH = 000073FD
RESULT = 0000000C
STRING = 00000008
TOPTAB = 00000003 R 02
VALTAB = 00000006 R 02

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes
ABS	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
YCVTATB	0000007A (122.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.06	00:00:01.45
Command processing	132	00:00:00.65	00:00:04.93
Pass 1	347	00:00:08.22	00:00:26.24
Symbol table sort	0	00:00:00.50	00:00:01.49
Pass 2	49	00:00:03.00	00:00:10.13
Symbol table output	11	00:00:00.10	00:00:00.30
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	578	00:00:12.55	00:00:44.56

The working set limit was 1500 pages.
36813 bytes (72 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 370 non-local and 9 local symbols.
2936 source lines were read in Pass 1, producing 21 object records in Pass 2.
134 pages of virtual memory were used to define 133 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	0
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	4

420 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:CVTATB/OBJ=OBJ\$:CVTATB MASD\$:[EMULAT.SRC]MISSING/UPDATE=(MASD\$:[EMULAT.ENH]MISSING)+MASD\$:[SYS.SRC]CVTATB/UPDATE=(MAS

0373

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY